Spring and EJB 3 Integration

Reza Rahman
Independent Consultant
Author, EJB 3 in Action
reza@rahmannet.net
EJB 3: EJB Reinvented

- Standards-based, vendor-agnostic server-side component development model
- Time-tested services for enterprise applications (e.g. transactions, security, persistence, messaging)
- Radical transformation of EJB 2.x
- 100% annotations, no XML deployment descriptor
- Intelligent defaulting, zero-configuration
- Powerful, easy-to-use, popular JPA
- EJB 3 adoption increasing steadily – despite the odds
- EJB 3.1 adds even more ease-of-use, features
Spring: Popular Java EE framework

- Rise as a response to EJB 2.x complexity, along with Hibernate
- Popular open-source middleware framework
- Dependency injection and AOP as core services
- Integration with standard (JPA, JMS, JDBC, JAX-WS) and non-standard tools (iBATIS, Quartz, Acegi)
- Extensible platform, easy to integrate third-party tools
- Integration with EJB
- Spring Application Platform en route to become Java EE 6 certified
EJB 3 and Spring Integration Scenarios

- Innovative, best-of-breed applications taking advantage of the strong-points of both technologies

- EJB Features
  - Standards-based, zero-configuration, secure, transactional, thread-safe components - great for the service tier
  - First-class JPA integration
  - Statefulness, passivation, extended persistence context
  - Message Driven Beans
  - Robust web services and remoting
  - Clustering, load-balancing, pooling and resource optimization
  - IDE and application server tooling support
EJB 3 and Spring Integration Scenarios

- **Spring Features**
  - Advanced dependency injection via Spring beans
  - Advanced AOP, including `@AspectJ` support
  - Spring JDBC integration
  - Spring JMS integration
  - iBATIS, Hibernate, JDO, TopLink DAOs
  - Quartz integration
  - Integration with other third-party or in-house frameworks

- These are just suggestions, not recipes

- The sky is the limit in terms of integration…
Integration Mechanics

- Spring can be used inside Java EE application server
- EJB 3 embeddable containers (OpenEJB, EasyBeans, Embedded JBoss, Embedded Glassfish) can be embedded onto Tomcat with Spring
- Anything in JNDI can be injected into Spring
- Spring natively supports Java EE @Resource and @EJB injection annotations for resources in JNDI
- XML based injection can be done via jee schema: `<jee:jndi-lookup/>` (no need to use `<jee:local-slsb/>` and `<jee:remote-slsb/>` for EJB 3)
Integration Mechanics

- Enabling Spring `@Autowired` annotation in EJB 3 beans via Interceptors
- Lookup from Spring application context using Spring-supplied EJB 2 base class
- Use EJB 3 natively inside Spring using Spring Pitchfork…or another solution 😊
The Tools in the Demo

- **Apache OpenEJB**
  - Most mature EJB 3 embedded container
  - EJB 3 implementation for Apache Geronimo and IBM WebSphere Community Edition
  - Supports EJB 3 deployments in WAR
  - Could also have used EasyBeans, Embedded JBoss or Embedded GlassFish

- **Spring 2.5**
  - Much better annotation support
  - XML schema based configuration
  - Improved JPA, JMS, JDBC, JSF and EJB integration
  - @AspectJ support
The Tools in the Demo

- **Tomcat**
  - Predominant lightweight web container
  - Great integration with OpenEJB
  - Could also have used Jetty, WebSphere, WebLogic, JBoss, GlassFish, Oracle

- **Eclipse**
  - Predominant Java IDE
  - Excellent Java EE 5 support including JSF, JPA and EJB 3
  - Spring IDE can be used with Eclipse
  - Could also have used NetBeans
The Demo!
Summary

- Create best-of-breed solutions leveraging the strong points of both EJB 3 and Spring
- Benefit from ease-of-use and vendor neutrality
- Take advantage of advanced AOP and DI as well as broad range of integration APIs
- Mix and match from broad range of integration points
- Can be done on Tomcat
- Can be done on Java EE servers
- How about much better native EJB 3 support on the Spring framework and application platform?
- Have fun! 😊
References

Shameless plug alert!